COMPARISON BETWEEN COUPLED LOCAL MINIMIZERS METHOD AND DIFFERENTIAL EVOLUTION ALGORITHM IN DYNAMIC DAMAGE DETECTION PROBLEMS

Loris Vincenzi [a], Guido De Roeck [b], Marco Savoia [c]

[a] DIMEC - University of Modena and Reggio Emilia
via Vignolese 905, 41125 Modena, Italy

[b] Department of Civil Engineering, Division of Structural Mechanics, Katholieke Universiteit Leuven
Kasteelpark Arenberg 40, B-3001 Heverlee, Belgium

[c] DICAM - Structural Engineering, University of Bologna
viale Risorgimento 2, 40136 Bologna, Italy

Abstract

In the present paper, a comparison is made between the Coupled Local Minimizers (CLM) method and the Differential Evolution (DE) algorithm to perform FE model updating for the damage detection in a cracked beam. CLM method is a gradient-based method with multiple local optimization runs. DE algorithm is a direct search approach which uses a population of solution vectors collecting the design parameters. Two benchmark examples of damage assessment are considered, i.e., beams under flexural vibrations with one crack and two cracks, with unknown position and depth. The effectiveness of the two methods to obtain the set of unknown parameters has been verified by performing a number of optimization processes starting from initial values of parameters selected randomly. Both exact and pseudo-experimental input data are used. A statistical analysis of the optimization results is presented. Both methods give results much better than the classical gradient optimization method. Better performances in term of speed rate and precision have been obtained by CLM when the number of identified parameters is limited. On the other hand, DE shows good efficiency when the number of parameters increases or in the case of pseudo-experimental input data.

Keywords: FE model updating, Coupled Local Minimizers, Differential Evolution, global optimization, dynamic tests, damage assessment.

1. Introduction

In several areas of civil and mechanical engineering, the maintenance and retrofitting of existing structures require the diagnostic identification of damages. To this purpose, dynamic and static non-destructive testing techniques have received great attention in the engineering community in last three decades [1-4]. In particular, the modal analysis techniques are considered as promising methods for the damage detection at the global scale level.

Several methods for damage detection and Structural Health Monitoring (SHM) are based on the idea of vibration-based damage detection, where changes in modal parameters, such as frequencies, mode shapes and modal damping, are used to detect changes in physical properties of the structure (see for instance [5-8]). Dynamic tests, using for instance ambient vibrations, provide for a much larger number of experimental data with respect to static tests and, as in the case of bridge structures, can be repeated during the structural life without interrupting the use of the structure, in order to identify the occurrence of a damage state [9-10]. Other damage detection methods exist [11-12] in which a structural model is not required [5,11]; when the localisation of the damage is necessary, an updating procedure of the Finite Element model of the structure (called FE model updating) can be performed, where the unknown model properties are modified in order to obtain numerical predictions as close as possible to a set of measured data [13-14].

Therefore, in model updating procedure, an optimization problem must be solved where the objective function to be minimized (the cost function) is defined by the distance between the modal parameters obtained from the experimental tests and those given by a

numerical model of the structure. Nevertheless, the objective function is often non-smooth or even discontinuous and may contain multiple local minima, so that very efficient optimization methods are required.

The success of the application of the FE model updating method depends on the selection of the unknown (i.e. updating) parameters, on the definition of the optimization problem and the mathematical capabilities of the algorithm adopted to find the minimum of the cost function. In fact, when modifications in geometry are admitted in the model updating, as in the case of identification of the position of a damage, the cost function to be minimized in the identification process may present local minima.

Conventional gradient-based methods (as Newton and Quasi-Newton Algorithms) [15] typically have an efficient convergence rate, but they may reach different local minima, depending on the selected starting point. They use a local second-order approximation of the cost function to generate an approximate quadratic model, which is used to estimate the minimum of the cost function. Nevertheless, it is well known that gradient-based methods often fail if the optimization problem is ill conditioned [15]. A low accuracy solution is also obtained when the objective function is flat close to the solution, i.e., in the case of low sensitivity of the objective function with respect to the variation of the optimization parameters. This circumstance often occurs when the number of optimization parameters is greater than two and/or when experimental measures are used as input data for the optimization problem.

Global optimization techniques, as global sensitivity-based methods and direct search approaches (genetic algorithms and evolution strategies) are often preferred because they

can avoid the convergence in a local minimum. As an example, genetic algorithms have been used in [16-17] for damage detection problems using the modal properties of the structure.

Among them, Coupled Local Minimizers method [18] and Evolution approaches [19-20] are considered very promising global optimization strategies.

In the present paper, global optimization techniques are used to perform the model updating of a damaged structure. In particular, the Coupled Local Minimizers method and the Differential Evolution Algorithm are used to detect position and depth of a localized damage, and their numerical performances are compared.

The Coupled Local Minimizers (CLM) method is a very efficient global gradient-based method originally proposed by Suykens [21-23]. Teughels [14,18] reformulated the original algorithm in the context of the FE model updating and applied it for the damage assessment by means of damage functions. The method adopts a number of search points and couples multiple local optimization runs in order to create interaction and information exchange between those points. A relatively fast convergence is maintained, because the updated set of search points is defined using gradient information. Nevertheless, the global minimum is usually reached, since multiple search points are used simultaneously [18].

The Differential Evolution (DE) method is a parallel direct search method where N different vectors collecting the unknown parameters of the system are used in the minimization process [24-25]. The vector population is chosen randomly at the beginning and by adding weighted differences between vectors obtained from the previous population in the subsequent steps. The DE algorithm has been used in [26-27] to perform dynamic

structural identification and in [28] for inverse analysis problems concerning the derivation of the parameters of a material constitutive laws from experimental data. In [29], a comparison between different types of evolutionary algorithms is reported.

In the present paper, the performances of the CLM method and the DE algorithm for the damage assessment of cracked beams through FE model updating are compared. The presence of cracks in a structural member introduces local flexibilities, so modifying its dynamic behaviour. The changes of dynamic characteristics (frequencies and mode shapes) can be measured and subsequently used for damage detection. The cracks are modelled according to Chondros et al. [30] by linear flexural springs, whose stiffness depends on the crack width. The comparison is performed with reference to two benchmark problems: two simply-supported beams under flexural vibrations, with one crack and two cracks to be detected (two and four identification parameters, respectively). The challenge is to find, by FE model updating, the crack location and width (the latter being related to the flexural spring stiffness used to model the crack).

The linear elastic cracked beam model is a very simplified theoretical case-study, the dynamic behaviour of a cracked beam being usually heavy non linear, varying also in positive and negative bending oscillations [31]. Nevertheless, even if simpler with respect to real cases, the two benchmarks considered are interesting, because they present more than one local minimum over the domain of definition of the identification parameters, and are then challenging for identification methods.

Two different sets of optimization problems are studied. In the first case, exact values of modal parameters (frequencies and modal shapes of the first two modes) are adopted as

input. In the second case, the optimization process is performed using pseudo-experimental data (the modal parameters affected by some error), obtained by adding to the exact values of the input data some statistic scattering. The effectiveness of the two algorithms to find the set of optimization parameters is then compared by performing a statistical analysis of the optimization results.

Very good results are obtained by both algorithms. CLM gives better performances in term of speed rate and precision when the number of optimization parameters is limited. On the other hand, DE shows better efficiency when the number of the parameters increases or in the case of pseudo-experimental data.

The classical gradient algorithm, with starting input values randomly selected, is also used for comparison. In this case, about 30 percent of tests failed for the one-crack problem, whereas in the two-crack problem very few simulations reached the correct solution.

2. Coupled Local Minimizers method

Consider the optimization problem defined as:

$$\textit{Find} \quad \mathbf{x} = [x_1, ..., x_i, ..., x_D] \quad \textit{such that} \quad H(\mathbf{x}) = \min \tag{1}$$

where **x** is the vector containing the *D* optimization parameters and $H(\mathbf{x})$ is the cost function to be minimized. In modal optimization problems, the cost function is typically defined as a function of the differences between measured (for example, from experimental tests) and calculated (from a numerical model, adopting a given set of optimization parameters) modal parameters (typically frequencies and mode shapes).

The Coupled Local Minimizer (CLM) algorithm [14, 21] is a global search method of minima of cost functions based on a cooperative search mechanism. It combines the fast convergence of the local gradient-based algorithms with the information exchange between different search points, typical of Genetic Algorithms.

In the CLM method, a set of q local minimizers is used in the optimization process:

$$\mathbf{X}^T = \left[\mathbf{x}_1{}^T, ..., \mathbf{x}_i{}^T, ..., \mathbf{x}_q{}^T\right] \in \Re^{1\times(D\cdot q)}$$

where each minimizer $\mathbf{x}_i$ contains the whole set of D optimization parameters. The population of local minimizers (the search points) is initially selected spread randomly over the search space. The cost function gradient information calculated for each search point directs the global search process. But instead of performing independent searches from each point (as in the case in multistart local optimization algorithms), the search points are coupled during the process by constraints that enforce the convergence to the same final point (see Fig. 1).

Hence, the objective function $H(\mathbf{x})$ in Eq. (1) is substituted by $f(\mathbf{X})$, being the sum of the values $H(\mathbf{x}_i)$ of the cost function corresponding to the local minimizers $\mathbf{x}_i$ (with i=1,…,q), i.e.:

$$f(\mathbf{X}) = \frac{1}{q}\sum_{i=1}^{q} H(\mathbf{x}_i) \qquad (2)$$

2.1.Augmented Lagrangian method

According to [18], the CLM algorithm is implemented with the Augmented Lagrangian method, where the local minimizers $\mathbf{x}_i$ (i=1,…,q) are coupled during the search process by

constraints enforcing the global search process to converge in the same final point.

A new objective function L_A is then constructed, by adding constraints to the objective function $f(\mathbf{X})$ to be minimized. The constrained minimization problem is then written as:

$$\textit{Find} \quad \mathbf{X}^T = \left[\mathbf{x}_1{}^T, ..., \mathbf{x}_i{}^T, ..., \mathbf{x}_q{}^T\right] \quad \textit{such that} \quad f(\mathbf{X}) = \min \tag{3}$$

subject to the synchronization constraints:

$$\mathbf{x}_i - \mathbf{x}_{i+1} = \mathbf{0} \qquad \text{for } i = 1,2,....,q \tag{4}$$

where q+1≡1.

According to the Augmented Lagrangian method for constrained minimization problems, the problem in Eq. (3, 4) is replaced by:

$$L_A(\mathbf{X}, \boldsymbol{\Lambda}) = \frac{\eta}{q}\sum_{i=1}^{q} H(\mathbf{x}_i) + \sum_{i=1}^{q}\langle \boldsymbol{\lambda}_i, (\mathbf{x}_i - \mathbf{x}_{i+1})\rangle + \frac{\gamma}{2}\sum_{i=1}^{q}\left\|\mathbf{x}_i - \mathbf{x}_{i+1}\right\|^2 \tag{5}$$

where a set $\boldsymbol{\Lambda}^T = \left[\boldsymbol{\lambda}_1{}^T, ..., \boldsymbol{\lambda}_i{}^T, ..., \boldsymbol{\lambda}_q{}^T\right]$ of q explicit Lagrangian multiplier vectors and a quadratic penalty term are added to the original objective function $f(\mathbf{X})$ [32]. Moreover, $\|\cdot\|$ denotes the Euclidean norm, γ is a penalty parameter, η a weight factor, and q+1≡1 as before. The second and the third term in Eq.(5) are called soft and hard constraints, respectively.

Differently from other algorithms (for example the quadratic penalty algorithm [15]), the Augmented Lagrangian method reduces the possibility of ill conditioning of the Hessian matrix $\nabla^2_{\mathbf{x}_i} L_A$ by the introduction of the explicit Lagrangian multipliers [32].

For a given k – *th* iteration, the set of vectors $\mathbf{x}_i$ and Lagrangian multiplier vectors $\boldsymbol{\lambda}_i$ minimizing the cost function in Eq.(5) is obtained iteratively with a two-step procedure.

- ***Step one:*** The Augmented Lagrangian Function L_A is minimized first with respect to the optimization variables collected in the vector **X**, for fixed values of the Lagrangian multipliers $\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_i, ..., \boldsymbol{\lambda}_q$ and γ (obtained in the previous *k*-1 step), i.e.:

$$\mathbf{0} = \nabla_{\mathbf{x}_i} L_A = \frac{\eta}{q} \nabla_{\mathbf{x}_i} H(\mathbf{x}_i) - (\boldsymbol{\lambda}_{i-1} + \gamma [\mathbf{x}_{i-1} - \mathbf{x}_i]) + (\boldsymbol{\lambda}_i + \gamma [\mathbf{x}_i - \mathbf{x}_{i+1}]) \quad \text{for } i = 1,\ldots,q \qquad (6)$$

so obtaining a set **X***.of pseudo-optimal vectors $\mathbf{x}_i^*$.

- ***Step Two :*** In order to satisfy the Karush-Kuhn-Tucker conditions (i.e. the first-order optimality condition to minimize the constrained problem), the values of the Lagrangian multipliers $\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_i, ..., \boldsymbol{\lambda}_q$ are updated by imposing the vanishing of the gradient of the Augmented Lagrangian Function L_A with respect to $\boldsymbol{\lambda}_i$:

$$\mathbf{0} = \nabla_{\boldsymbol{\lambda}_i} L_A \quad \rightarrow \quad \boldsymbol{\lambda}_i^* = \boldsymbol{\lambda}_i + \gamma [\mathbf{x}_i - \mathbf{x}_{i+1}] \quad \text{for } i = 1,\ldots,q \qquad (7)$$

The two-step iterative process continues until the optimal variables $\mathbf{X}^T = [\mathbf{x}_1^T, ..., \mathbf{x}_i^T, ..., \mathbf{x}_q^T]$ and the optimal Lagrangian multipliers $\boldsymbol{\Lambda}^T = [\boldsymbol{\lambda}_1^T, ..., \boldsymbol{\lambda}_i^T, ..., \boldsymbol{\lambda}_q^T]$ are found, up to a prescribed tolerance for convergence.

2.2.Trust Region Newton Method

In [21, 22, 32], a steepest descent method [15] is used to minimize L_A with respect to $\mathbf{x}_i$ (step one of the procedure in Section 2.1). According to [14], a standard Trust Region Newton method is used here in order to improve the convergence speed and to obtain a robust optimization process. To apply the Trust Region Method, a truncated Taylor series of the Augmented Lagrangian function is defined, i.e.:

$$m(\mathbf{p}) = L_A + \left[\nabla_{\mathbf{X}} L_A\right]^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T \left[\nabla^2_{\mathbf{X}} L_A\right]\mathbf{p} \tag{8}$$

where $\mathbf{p} = \Delta\mathbf{X}$. Eq. (8) defines a quadratic approximation $m(\mathbf{p})$ of L_A in the neighbourhood of $\mathbf{X}$ defined by the constraint $\|\mathbf{p}\| \leq \Delta$, where $\Delta > 0$ is the Trust Region radius. In other words, $\mathbf{p}$ represents a step-vector $\Delta\mathbf{X}$ restricted to the Trust Region. Assuming the relative independence of local minimizers, the Gradient and the Hessian matrices ($\nabla_{\mathbf{X}} L_A$ and $\nabla^2_{\mathbf{X}} L_A$) in Eq. (8) can be written as a function of the Gradient and the Hessian matrices computed with respect to the individual search points, i.e. (for $i = 1,\ldots q$):

$$\nabla_{\mathbf{x}_i} L_A = \frac{\eta}{q}\nabla_{\mathbf{x}_i} H(\mathbf{x}_i) - (\boldsymbol{\lambda}_{i-1} + \gamma[\mathbf{x}_{i-1} - \mathbf{x}_i]) + (\boldsymbol{\lambda}_i + \gamma[\mathbf{x}_i - \mathbf{x}_{i+1}]) \tag{9}$$

$$\nabla^2_{\mathbf{x}_i} L_A = \frac{\eta}{q}\nabla^2_{\mathbf{x}_i} H(\mathbf{x}_i) + 2\gamma\mathbf{I} \tag{10}$$

$$\nabla^2_{\mathbf{x}_i,\mathbf{x}_{i-1}} L_A = -\gamma\mathbf{I} \tag{11}$$

$$\nabla^2_{\mathbf{x}_i,\mathbf{x}_{i+1}} L_A = -\gamma\mathbf{I} \tag{12}$$

where $\nabla^2_{\mathbf{x}_i,\mathbf{x}_j} L_A = \mathbf{0}$ otherwise and $\mathbf{I}$ is the identity matrix.

The vector $\mathbf{p}_k$ is obtained by minimizing L_A with respect to $\mathbf{X}$ with a second iterative procedure, approximating L_A by Eq. (8) and taking the Trust Region constraint into account. The size of the region Δ is modified at each subiteration, depending on how well the second order approximation function $m(\mathbf{p})$ in Eq. (8) agrees with L_A. At the general j-th sub-step, $\mathbf{p}_k^{(j)}$ is obtained from Eq. (8). Then, both exact value $L_A\left(\mathbf{X} + \mathbf{p}_k^{(j)}\right)$ and that predicted by the quadratic model through Eq. (8), i.e., $m_k(\mathbf{p}_k^{(j)})$ are computed at the new point. The ratio of actual to predicted reduction of the cost function is finally computed:

$$\rho(\mathbf{p}_k^{(j)}) = \frac{L_A(\mathbf{X}) - L_A\left(\mathbf{X} + \mathbf{p}_k^{(j)}\right)}{m_k(\mathbf{0}) - m_k(\mathbf{p}_k^{(j)})}$$

If $\rho(\mathbf{p}_k^{(j)})$ is close to 1, then the quadratic approximation in Eq. (8) is a good predictor of the actual cost function, and the Trust Region can be increased in size by increasing the radius Δ. On the contrary, if $\rho(\mathbf{p}_k^{(j)})$ is far from one, the Trust Region is decreased in size. If far from a threshold value, the step is rejected and recomputed.

In order to verify that the local minimizers converge in the same final point, the convergence rule is based on the distance between their individual components (j=1,…,D), i.e.:

$$\left|\frac{x_{ji} - x_{ji+1}}{x_{ji}}\right| < VTR \quad \text{for } j = 1,...,D;\ i = 1,...,q \tag{13}$$

where *VTR* is the prescribed precision constant.

Since a Newton-based method is used (see Section 2.1), the quality of the solution depends on a number of parameters selected a-priori, i.e., on the number q of local minimizers, the penalty parameter γ and the weight factor η (see Eq. (4)).

The shape complexity of the original cost function $H(\mathbf{x})$ and the possible presence of local minima must be considered when assigning the number q of local minimizers: for regular functions, few search points are needed to obtain a good performance, whereas a higher number is needed to reach the global minimum in the presence of local minima of the cost function.

The values of the penalty parameter γ and the weight factor η must be also prescribed a-priori at the beginning of the analysis. The difficulty of selection of the values of these

parameters is typical in global search methods but, in this case, an incorrect parameter definition can determine the failure of the optimization process. With reference to Eq.(4), by increasing γ with respect to η, more weight is given to the constraints between the search points. Consequently, the convergence rate is improved, but the convergence to local minima may occur. On the other hand, a low value of γ allows for a more complete exploration of the search domain, but the convergence speed decreases.

A detailed description of the CLM method including implementation, numerical tests on analytical cost functions and some applications to FE model updating and damage detection problems can be found in [18].

3. Differential Evolution Algorithm

A random direct search method is a strategy based on the simultaneous search of a number *NP* of parameter vectors initially generated randomly in order to explore the whole space of definition of them. Then, variations of the parameter vectors are generated in order to minimize the objective function. In this step, a degree of randomness is still introduced in the procedure. Once a variation is generated, a decision must be made whether or not to accept the new parameter vectors. A new vector of parameters is accepted only if it reduces the value of the objective function. If some vectors reach local minima, they can be excluded because they are associated with higher values of the cost function to be minimized.

In the present study, the Differential Evolution (*DE*) algorithm is used [24]. It is a heuristic direct search approach where *NP* vectors:

$$\mathbf{x}_{i,G}, \quad i = 1,2,...,NP$$

are used, where the subscript *G* indicates the *G*-th generation of the parameter vectors, called *population*. Each vectors $\mathbf{x}_{i,G}$ contains a number *D* of optimization parameters. The number *NP* of vectors of the population is kept constant during the minimization process. The population is changed up to the attainment of the required convergence limits (see Section 3.5).

The algorithmic scheme of the DE algorithm is shown in Fig. 2*a*. The initial population is first chosen randomly. Then, in the *Mutation* and *Crossover* operations, DE generates new vectors to be compared with members of the previous population. If one newly generated vector gives a lower value of the objective function than that belonging to the old population, it replaces the old vector (*Selection* operation). These operations are described with more details in the following.

3.1.Mutation

For each vector $\mathbf{x}_{i,G}$ of the *G*-th population, a mutant vector $\mathbf{v}_{i,G}$ is generated by adding to $\mathbf{x}_{i,G}$ a contribution given by the difference between two other vectors of the same population.

Three different combination strategies can be used during the mutation process: the "random" combination, the "best" combination and an intermediate combination called "best-to-rand".

According to Storn and Price [24], in the random combination, the *mutant vector* $\mathbf{v}_{i,G}$ is generated according to the expression:

$$\mathbf{v}_{i,G+1} = \mathbf{x}_{r_1,G} + F \cdot (\mathbf{x}_{r_2,G} - \mathbf{x}_{r_3,G}), \quad (14)$$

where $r_1, r_2, r_3 \in \{1,2,...,NP\}$ are mutually different integer numbers. Moreover, *F* is a positive constant (scale parameter) controlling the amplitude of the mutation. The scale parameter *F* is taken < 2.

The "Best" combination is similar to the random combination, but the mutant vector is defined from the equation:

$$\mathbf{v}_{i,G+1} = \mathbf{x}_{best,G} + F \cdot (\mathbf{x}_{r_1,G} - \mathbf{x}_{r_2,G}), \quad (15)$$

where $\mathbf{x}_{best,G}$ is the vector giving the minimum value of the objective function of the previous G-*th* population.

Finally, in the "best-to-rand" combination, the mutant vector is generated according to the expression:

$$\mathbf{v}_{i,G+1} = \mathbf{x}_{i,G} + F \cdot (\mathbf{x}_{best,G} - \mathbf{x}_{i,G}) + F \cdot (\mathbf{x}_{r_1,G} - \mathbf{x}_{r_2,G}). \quad (16)$$

The effectiveness of one method depends on the regularity of the objective function. For regular functions with only one (global) minimum, the "best" combination converges more rapidly because the best vector obtained from the previous generation is taken as the basic vector for the subsequent vector population. In the presence of local minima, the "random" or "best-to-rand" combinations must be preferred, because the convergence to local minima can be avoided.

3.2.Crossover

In order to increase the diversity of the vectors, the crossover process is introduced in the DE algorithm.

The *trial vector* $\mathbf{u}_{i,G+1}$ is obtained by randomly exchanging the values of the optimization parameters between the original vectors of the population $\mathbf{x}_{i,G}$ and those of the mutant population $\mathbf{v}_{i,G+1}$, i.e.:

$$\mathbf{u}_{i,G+1} = (u_{1i,G+1}, u_{2i,G+1}, ..., u_{Di,G+1}),$$

where:

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} & if\ \ rand(j) \le CR \\ x_{ji,G} & if\ \ rand(j) > CR \end{cases} \tag{17}$$

In Eq. (17), $j = 1,2,...,D$, where D is the number of optimization parameters, and u_{ji} is the j-th component of the vector $\mathbf{u}_i$.

Moreover, *rand*(j) is the j-th value of a vector of uniformly distributed random numbers, and CR is the crossover constant, with $0 < CR < 1$. Constant CR indicates the percentage of mutations considered in the trial vector.

3.3.Selection

In order to decide if a vector $\mathbf{u}_i$ may be the element of the new population (generation G+1), each vector $\mathbf{u}_{i,G+1}$ is compared with the previous corresponding vector $\mathbf{x}_{i,G}$. If the vector $\mathbf{u}_{i,G+1}$ gives a value of the objective function H smaller than $\mathbf{x}_{i,G}$, $\mathbf{u}_{i,G+1}$ is selected as the new vector of population G+1; otherwise, the old vector $\mathbf{x}_{i,G}$ is retained:

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1} & \text{if } H(\mathbf{u}_{i,G+1}) < H(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G} & \text{if } H(\mathbf{u}_{i,G+1}) \ge H(\mathbf{x}_{i,G}) \end{cases} \tag{18}$$

with $i = 1,2,...,NP$.

3.4.Bound constraints

Usually, in engineering applications, the optimization parameters are constrained to belong in given intervals, i.e.:

$$x_{ji,G} \in \left\lfloor x_{ji,\min}, x_{ji,\max} \right\rfloor ,$$

where $j = 1,2,...,D$.

Introducing the bound constraints for the optimization parameters is useful in order to restrain the analysis to ranges of unknown parameters which are meaningful from the physical point of view (see Section 5). To this purpose, a projection algorithm is introduced. After the mutant operation, if a vector out of range is obtained, its projection on the prescribed interval of parameters is considered (see Ref. [27] for details).

3.5.The convergence rule

In the convergence rule, the values of the objective function obtained from the vectors of the population $G+1$ are compared. The vectors are ordered as a function of the values of the objective function as:

$$\tilde{\mathbf{x}}_{1,G+1} \prec \tilde{\mathbf{x}}_{2,G+1} \prec ... \prec \tilde{\mathbf{x}}_{NP,G+1},$$

such that:

$$H\left(\tilde{\mathbf{x}}_{1,G+1}\right) < H\left(\tilde{\mathbf{x}}_{2,G+1}\right) < ... < H\left(\tilde{\mathbf{x}}_{NP,G+1}\right).$$

The convergence rule is defined by selecting the number *NC* of vectors to be controlled. The convergence test is then written as a function of the values of the objective function corresponding to the controlled vectors:

$$\Delta_i^H = \frac{\left| H(\tilde{\mathbf{x}}_{i,G+1}) - H(\tilde{\mathbf{x}}_{i+1,G+1}) \right|}{\left| H(\tilde{\mathbf{x}}_{i,G+1}) \right|} < VTR_1 \,, \tag{19}$$

where $i = 1,..., NC$ and VTR_1 is the prescribed precision.

The convergence test in Eq. (19) can be not sufficient when the objective function has a low gradient close to the minimum. For this reason, a second convergence test is introduced, requiring also the relative distance between the components of the first *NC* vectors be small, i.e.:

$$\Delta_{ij}^x = \frac{\left| \tilde{x}_{ji,G+1} - \tilde{x}_{ji+1,G+1} \right|}{\left| \tilde{x}_{ji,G+1} \right|} < VTR_2 \,. \tag{20}$$

with VTR_2 being the prescribed precision.

4. Damage detection in a cracked beam by FE model updating

4.1.The objective function

In dynamic optimization problems, the unknown properties of a numerical model are adjusted in order to obtain the numerical predictions of the modal parameters as close as possible to the measured values [27, 33-35]. The unknown parameters are then obtained by minimizing the cost function, defined by the distance between the modal parameters obtained from the experimental tests and those given by a numerical FE model of the structure. During the updating process, natural frequencies and mode shapes of the model can not appear in the same order as the reference ones. For this reason, the reference and the numerical modes are first coupled by using the Modal Assurance Criterion (*MAC*) [33]. Then, the cost function can be defined by the following least square expression:

$$H(\mathbf{x}) = \frac{1}{2}\left\| \begin{matrix} \mathbf{r}_f(\mathbf{x}) \\ \mathbf{r}_s(\mathbf{x}) \end{matrix} \right\|^2 \tag{21}$$

where $\mathbf{r}_f(\mathbf{x})$ and $\mathbf{r}_s(\mathbf{x})$ are the residual vectors and collect the differences between numerical and experimental modal parameters in terms of eigenfrequencies and mode shapes, respectively. Their components are:

$$r_{f,h}(\mathbf{x}) = \frac{\omega_h(\mathbf{x}) - \overline{\omega}_h}{\overline{\omega}_h} \tag{22}$$

$$r_{s,hk}(\mathbf{x}) = \frac{\varphi_{hk}(\mathbf{x})}{\varphi_{hr}(\mathbf{x})} - \frac{\overline{\varphi}_{hk}}{\overline{\varphi}_{hr}} \tag{23}$$

where ω_h and $\boldsymbol{\varphi}_h$ are the undamped eigenfrequency and corresponding eigenvector for the *h*-th mode. In particular, the modal parameters obtained from the tests are denoted as $(\overline{\omega}_h, \overline{\boldsymbol{\varphi}}_h)$, whereas $(\omega_h(\mathbf{x}), \boldsymbol{\varphi}_h(\mathbf{x}))$ are those obtained from the numerical model adopting the set **x** of identification parameters. Moreover, in Eqs.(22), (23), $h = 1,...,NM$, where *NM* is the number of modes considered, and subscripts *k* and *r* denote the general and a reference degree of freedom, the latter adopted for normalization purpose.

Since the CLM is a sensitivity-based method, the Gradient and the Hessian matrix can be analytically obtained from the partial derivatives of the residuals $\mathbf{r}_f(\mathbf{x})$ and $\mathbf{r}_s(\mathbf{x})$ with respect to **x**. The modal sensitivities are calculated using the formulas by Fox and Kapoor [36-37]. A detailed description is presented in [18].

4.2.Cracked beam model

The presence of a crack in a structural member introduces a local flexibility that modifies its dynamic behaviour. Several methods exist to perform the damage detection by

measuring the change of the modal parameters from the undamaged to the damaged configuration (see for instance [38]). Some methods perform the identification of the damage location and depth separately in a two-stage algorithm, the first problem being the more difficult task, typically solved by a FE model updating procedure [39].

In Ref. [27], fracture mechanics has been used to model the crack as a discrete elastic flexural spring. It is assumed that the crack in a structural element is open and remains open during the vibrations. The spring stiffness *K* depends on the crack depth and can be evaluated by means of the following formula [30]:

$$K = \frac{EI}{6\pi h(1-\nu^2)\varepsilon} \tag{24}$$

where *h* is the beam height, *I* is the second moment of inertia, *E* is the Young's modulus of elasticity and ν is the Poisson ratio. Moreover, for a rectangular beam, the parameter ε can be evaluated as a function of the crack depth *a* by means of the following approximate formula:

$$\begin{aligned}\varepsilon = {} & 0.6272\left(\frac{a}{h}\right)^2 - 1.04533\left(\frac{a}{h}\right)^3 + 4.5948\left(\frac{a}{h}\right)^4 - 9.9736\left(\frac{a}{h}\right)^5 + 20.2948\left(\frac{a}{h}\right)^6 \\ & - 33.0351\left(\frac{a}{h}\right)^7 + 47.1063\left(\frac{a}{h}\right)^8 - 40.7556\left(\frac{a}{h}\right)^9 + 19.6\left(\frac{a}{h}\right)^{10}\end{aligned} \tag{25}$$

4.3.The case studies

In order to test the numerical performances of the Coupled Local Minimizers method and the Differential Evolution algorithm, the damage assessment of a simply supported, cracked, aluminium beam is performed by FE model updating. The beam has a rectangular

cross-section 6×25.4 (b×h) mm and length of 235 mm. The material properties are: Young's modulus $E = 7.2 \cdot 10^4$ MPa, density 2800 kg/m^3 and Poisson's ratio 0.35. The reference solution and the numerical solutions for a given set of identification parameters are obtained by ANSYS [40], with 30 linear elastic beam elements and the cracks modelled by elastic springs. Two different case studies have been considered: the beam with one and two flexural cracks, with unknown positions and depths.

5. Results of the damage assessment problem

5.1. One-crack detection problem (two-parameter optimization)

The first case refers to a beam with one crack with depth 9.7 mm and placed at 1/3 of the length l (see Fig. 3*a*). The identification problem has two unknown parameters, the location and the stiffness of the spring modelling the crack according to Eq. (24). First of all, a modal analysis is performed to obtain the first two natural frequencies and mode shapes (see Table 1 and Fig. 3*a*). These results are the target or "Reference solution". The search space of the two updating variables, stiffness and position, is limited to the intervals $X_1 = K$ =] 0, 28000 [Nm, $X_2 = y$ =] 0, 235 [mm. For numerical convenience, the normalized value x_j of the identification parameter X_j is defined as:

$$x_j = \frac{X_j - X_{jS}}{X_{jD}} \quad \Rightarrow \; X_j = x_j X_{jD} + X_{jS}\,, \tag{26}$$

where:

$$X_{jS} = \frac{X_{j2} + X_{j1}}{2}, \quad X_{jD} = \frac{X_{j2} - X_{j1}}{2}$$

are the mean value and width of the interval of variation of X_j, so that the normalized

parameter varies in the range] -1, 1 [.

The cost function is defined according to Eqs. (21-23), where the number of modes is $NM = 2$. The 3D plot and the contour lines of the objective function are shown in Figs. 4*a*, *b*.

In order to conduct a meaningful comparison between the performances of the two methods, a preliminary study has been performed where the optimal values of the parameters required by both methods have been calibrated to find rapidly the global minimum and to reach similar precision. For the criteria to be used to define the algorithm parameters of the two methods, the reader is referred to Refs. [21, 24].

As far as the CLM method is concerned, 4 local minimizers are used; moreover, $\gamma = 10$ and $\eta = 0.5$ are set for the penalty parameter and the weight factor. The convergence test reported in Eq. (13) is adopted, with $D = 2$, $q = 4$ and $VTR = 0.01$.

5.1.1. Exact input data

In the first numerical tests, the exact values of frequencies and eigenvectors are used as input data, i.e., no measurement errors are considered. The first two frequencies and modes, with 15 positions for the definition of the deformed shape, are used.

100 identification tests have been performed by selecting randomly different sets of 4 starting search points. Random values are also selected for the initial values of the Lagrangian parameters $\mathbf{\Lambda}$.

The same number of identification tests is then performed by adopting the DE algorithm, with random values for the initial search points. The *Best combination* strategy

is adopted in all numerical tests and $F = 0.6$ is set. The convergence test is defined using the relative errors between the three best members of the same population (those giving the minimum values of the cost function), according to Eqs. (18, 19). The values adopted for the convergence constants are $VTR_1 = 0.01$ and $VTR_2 = 0.02$. The number of vectors for each population is $NP = 10$, selected randomly at the beginning of the identification process.

The first comparison in terms of algorithm performances is made with reference to the number of failed tests. As far as the CLM method is concerned, 1 test over 100 failed; in this case, the solution point reached the boundary. This is due to the low number of minimizers adopted and the random choice of their initial values: if the points are not well scattered, a local minimum (in this case on the boundary) can be found. One test failed also with DE, not because of the wrong localization of the identification parameters, but because of the low accuracy (error greater than 2%) on the spring stiffness identification after the prescribed maximum number of iterations. A classical gradient – based optimization algorithm has been also used, with starting point selected randomly: in this case, 30 % of the tests failed to reach the convergence.

The statistical analysis of the identification results obtained by CLM and DE algorithms is reported in Table 2. The mean values of the updating parameters (damage position and spring stiffness) and the number of iterations required for the convergence are reported. The range of results and coefficient of variation (standard deviation divided by mean value, C.V.) are also given.

It is worth noting that very good results are obtained using both CLM and DE

algorithms: the mean values of the identification parameters obtained from 100 numerical tests are very close to the target values (the reference solution).

For the CLM method, the errors on location (0.30 %) and spring stiffness (0.25%) are comparable; the coefficients of variations are also very similar (0.39 % vs. 0.28 %). With the DE algorithm, very good results are obtained as far as the crack position is concerned: the relative error is 0.11% and the coefficient of variation is less than 0.1%. Nevertheless, the identified stiffness values are more spread than in the case of CLM method, with a coefficient of variation of about 0.6%. The different performances are due to the shape of the objective function, which is almost insensitive to the spring stiffness variation close to the solution point (see Fig. 5). The coupling constraints introduced in the CLM method enforce the local minimizers close to each other in the solution (see Fig. 5). On the contrary, adopting the DE, at the end of the procedure all search points tend to be located along a line, with small variation in the crack position and higher dispersion of stiffness results.

As shown in Table 2, by using the DE algorithm the mean number of iterations is about 23, whereas it is lower (about 14) adopting the CLM.

The number of cost function evaluations is also given in order to compare the convergence speed rate of the two methods. For each iteration, the CLM method uses 4 search points; every point needs three evaluations of the cost function (two are used to obtain the sensitivity matrix). The DE algorithm has the disadvantage of requiring a larger number of function evaluations (10 search points for each iteration in this example), because it is based on a statistical searching, but it does not need any gradient information.

Hence, the number of cost function evaluations required by the DE algorithm is only 20% greater than using the CLM method.

The identification tests are then repeated by adopting 5 positions only for the definition of the shape functions of the two vibration modes. The results are reported in Table 3. The errors are still small, even if of course greater than in the previous case (15 positions). Using DE and CLM, the mean error over the 100 identification tests is about 0.5 % and 3 %, respectively, with C.V. less than 3 %. Moreover, the number of measurement positions has no effect on the number of cost function evaluations required for convergence.

5.1.2. Pseudo-experimental input data

In order to test the robustness of the two algorithms, the optimization tests have been performed also using pseudo-experimental data, obtained by adding some statistic scattering to the exact values of the input data. To this purpose, the pseudo-experimental data are obtained by multiplying the exact values of frequencies and components of mode eigenvectors by random uncorrelated coefficients, extracted from normal probability distributions with unit mean value and prescribed C.V.. According to typical variations of modal parameters obtained from modal experimental tests, the C.V.s of distributions are set equal to 1 percent for the frequencies and 5 percent for the eigenvector components.

Also in this case, gradient – based algorithms give very bad performances in parameter identification, being the number of failed tests about 30 % of the total number of simulations. For this reason, only the two global search methods are considered in the following tests.

The statistical analysis of the identification results obtained by CLM and DE algorithms

is reported in Tables 4, 5 for tests conducted with 15 and 5 measurement positions along the beam, respectively. For each individual test, the scatter with respect to the reference values to be identified is of course due to both the error introduced in the input data and the numerical error of the algorithm. Using 15 measurement positions, the mean error over 100 identification tests is comparable with that of results obtained with exact input data (see Table 2). The C.V. of the results is slightly greater, but always smaller than 3 %. The results are worse using 5 measurement positions. Adopting DE, the mean errors are still under 1% and the C.V.s are double with respect to the previous test. Adopting CLM, the results are affected by a significant error: for instance, the mean value of the spring stiffness predicted over 100 tests is overestimated by 7.5%.

5.2.Two-crack detection problem (four-parameter identification)

In order to verify the ability of the two algorithms to find the correct solution also with a larger number of updating parameters, a two-crack detection problem has been considered. Four parameters must be identified, i.e., position and depth of the two cracks. The identification problem of multiple cracks along the beam is much more severe than the one-crack detection problem, because the frequency variation due to the presence of additional cracks is usually very small [30, 41].

As in the case of the one-crack detection problem, the identification parameters are normalized and the solution is searched with parameters defined in the interval]+1,-1[. In the reference solution to be identified, the two cracks are located at 1/3 l and 5/8 l, and the stiffness values are set K_1 = 15680 Nm, K_2 = 11900 Nm.

Ten local minimizers are used for the CLM algorithm. A penalty factor $\gamma = 25$ has been

adopted to avoid local minima and $\eta = 25$ is set. In order to obtain good results, the prescribed precision $VTR = 0.005$ is set in Eq. (12). For the DE algorithm, the number of search points is set to 25. The same convergence rules as in the one crack detection problem are adopted.

The comparison between the results obtained with the two different algorithms is reported in Table 6. As shown, results with accuracy comparable with that of the one crack detection problem are obtained for the DE algorithm. The mean value of the number of cost function evaluations is about 750. No tests over 100 simulations failed.

The results are very close to the exact values for both cracks. For the stiffnesses, the results show a higher coefficient of variation, due to the lower sensitivity of the modal parameters. This is agreement with Ref. [42], where the accuracy in predicting the crack depth was lower than for the crack location also in the case of identification from dynamic experimental tests.

The CLM method performed similarly in term of speed rate (mean number of evaluations is about 780) but with a higher C.V. of identification results for both positions and stiffnesses. Moreover, about 7% of the tests failed. It is worth noting that the penalty factor γ and the number of CLMs must be carefully chosen to avoid local minima or an excessive number of iterations. If a number of search points lower than 10 is used, the solution attained mat significantly depend on the selected starting points. For example, with 6 or 8 points instead of 10, the speed rate is improved (the mean values of cost function evaluations are about 400 and 590, respectively). Nevertheless, the number of failed tests is more than 20% in the first case and about 15% when using 8 minimizers. A number of

CLMs higher than 10 strongly increases the computational effort but the number of failed tests remains substantially unchanged.

As far as the DE is concerned, the number of search points and the parameter values of the algorithm modify the convergence speed rate but the global minimum is usually reached. Moreover, if a lower number of points is adopted, a higher number of iterations is usually required, without significant variation of the total number of cost function evaluations (about 750).

As far as the gradient – based algorithm is concerned, starting from random input data the solution has been rarely reached. This result confirms that gradient-based methods usually fail when the number of identification parameters is greater than two, due to the presence of low sensitivity regions of the objective function with respect to the optimization parameters [15].

The identification tests have been repeated using pseudo-experimental input data for frequencies and mode shapes. The results reported in Table 7 clearly show the better performances of DE over CLM algorithm. First of all, the C.V. of results is significantly smaller: less than 1 % for the crack position and 3 – 6 % for the spring stiffness, with respect to C.V.s equal to 4-5% and 20-30 %, respectively, of the results obtained by CLM. As shown in Fig 6, the worse performances can be due to the presence of local minima or a very small sensitivity of the cost function with respect to some identification parameters.

Moreover, the mean number of cost function evaluations required by DE is analogous as in the case of exact input data, whereas, using CLM, it is 5 times greater when pseudo-experimental input data are adopted.

5.3.Identification of a single-cracked beam by using a two-crack model

The results reported in Sections 5.1 and 5.2 have been obtained with the assumption that the number of cracks is known. In this section, a single-cracked beam (y = 78 mm, K = 15680 Nm) is identified by using a FE model where the presence of two cracks ia assumed. For better clarity, the "reference solution" is the single-cracked beam, whereas the "detection model" is the FE model with two springs.

For both CLM and DE methods, the number of search points, the values of algorithm parameters and prescribed precisions are the same adopted in Section 5.2.

As for the position detection, the minimum of the cost function is obtained when the two springs are located in the same position. The results obtained (100 identification tests) show that both DE algorithm and CLM method are able to detect the damage location; errors and C.V.s are very similar to those obtained for the two-crack beam problem, with maximum error smaller than 4%.

On the contrary, due to the different theoretical approaches of DE and CLM methods, different results are obtained as far as the identification of the crack depth is concerned.

The present problem is of course ill-posed, because no unique solution for the identification problem exists. In fact, the local flexibility introduced in the reference solution by the presence of the single spring can be obtained in the detection model by the sum of two local flexibilities. For this reason, ∞^1 solutions giving the minimum value of the cost function exist.

In the CLM method, the local optimizers are coupled during the search process by constraints enforcing the global search process to converge in the same final point. Hence,

the algorithm converges to a solution with different values of the two stiffnesses K_1, K_2, depending on the set of initial points adopted, but the results in term of total stiffness $K = K_1 + K_2$ show a precision comparable to the two-crack detection problem.

Using the DE algorithm, the search points attain vanishing values of the cost function after few iterations. Nevertheless, these points have different values of two spring stiffnesses, distributed over the subspace $K_1 + K_2 = K$, with K being the stiffness value of the reference solution (one crack). Hence, in order to satisfy the convergence rule (see Section 3.5), the test on the distance between vectors should be relaxed, because no constraints allowing the search points to group together as in CLM method are present.

6. Concluding remarks

In the present paper, two different techniques for FE model updating of a cracked beam are adopted. The cost function to be minimized is the distance between the values of the modal properties (frequencies and modal shapes), adopted as input data, and those obtained adopting a given set of unknown parameters, defining damage position and depth. In particular, the performances of the Coupled Local Minimizers (CLM) method and the Differential Evolution (DE) algorithm are compared. Analogous levels of convergence limits are set, and number of iterations and cost function evaluations required to attain it are compared.

The problem considered is the identification of position and width of one and two cracks in a beam under flexural vibrations. This problem presents more than one local minimum over the domain of definition of the identification parameters. Both exact and

pseudo-experimental input data, the latter simulating the presence of experimental errors, are used. For each identification problem, 100 simulation tests have been performed, and a statistical analysis of results has been performed. Very good results have been obtained with both Differential Evolution Algorithm and Coupled Local Minimizers Method. CLM method gives better performances in term of speed rate and precision when the number of identified parameters is limited. On the other hand, DE shows very good efficiency when the number of parameters increases, and when pseudo-experimental values are adopted as input data. Classical gradient-based algorithms with random starting point are found to be not appropriate for these identification problems, due to the presence of local minima: 30 percent of tests failed to reach the minimum for the one-crack beam, and very few tests reached the correct solution for the two-crack beam.

In a further study, the performances of the optimization algorithms will be compared with reference to model updating problems based on real experimental tests.

Acknowledgements

The study was conducted when Dr. Loris Vincenzi was visiting student at the Department of Civil Engineering - Katholieke Universiteit Leuven, Belgium. The paper has been completed when Marco Savoia was a fellow of the Italian Academy for Advanced Studies in America.

The financial support of the Department of Civil Protection (Reluis 2005 Grant – Task 7: Techniques for isolation and control of structures and infrastructures) is gratefully acknowledged.

References

[1] R.D. Adams, P. Cawley, C.J. Pye, B.J. Stone, A vibration technique for non-destructively assessing the integrity of structures. *J.Mech. Eng. Sci*. 20(2) (1978) 93–100.

[2] G. Hearn, R.B. Testa, Modal analysis for damage detection in structures. *J. Struct. Eng.* 117(10) (1991) 3042–3063.

[3] D. Capecchi, F. Vestroni, Monitoring of structural systems by using frequency data. *Earth. Eng. Struct. Dyn*. 28 (1999) 447–461.

[4] S. Caddemi, A. Morassi, Crack detection in elastic beams by static measurements. *Journal of Solids and Structures* 44 (2007) 5301-5315.

[5] S.W. Doebling, C.R. Farrar, M.B. Prime, D.W. Shevitz. Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review. *Los Alamos National Laboratory Report LA-13070-MS*, 1996;

[6] A.Teughels, J. Maeck, G., De Roeck, Damage assessment by FE model updating using damage functions. *Computers and Structures* 80(25) (2002) 1869-1879.

[7] J. Wang, P. Qiao, On irregularity-based damage detection method for cracked beams. *Journal of Solids and Structures* 45 (2008) 688-704.

[8] A. Deraemaeker, A. Preumont, E. Reynders, G. De Roeck, J. Kullaa, V. Lamsa, K. Worden, G. Manson, R. Barthorpe, E. Papatheou, P. Kudela, P. Malinowski, W. Ostachowicz, T. Wandowski. Vibration-based structural health monitoring using large sensor networks. *Smart Structures and Systems* 6(3) (2010) 335-347.

[9] L. Garibaldi, S. Marchesiello, E. Bonisoli, Identification and up-dating over the Z24 benchmark. *Mechanical Systems and Signal Processing* 17(1) (2003) 153–161.

[10] B. Peeters, G. De Roeck, Reference-based stochastic subspace identification for output-only modal analysis. *Mechanical System and Signal Process* 6(3) (1999) 855-868.

[11] H. Sohn, C.R. Farrar. Damage Diagnosis Using Time Series Analysis of Vibration Signals, *Smart Materials and Structures*, 10(3) (2001) 446-451

[12] K.K. Naira, A.S. Kiremidjianb, K.H. Lawb. Time series-based damage detection and localization algorithm with application to the ASCE benchmark structure. *Journal of Sound and Vibration* 2006; 291: 349-368.

[13] M.I. Friswell, J.E. Mottershead, *Finite Element model updating in structural dynamics*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[14] A. Teughels, G. De Roeck, J.A.K. Suykens, Global optimization by coupled local minimizers and its application to FE model updating. *Computers and Structures* 81(24) (2002) 2337-2351.

[15] G.N. Vanderplaats, *Numerical Optimization Techniques for Engineering Design*. McGraw-Hill, 1984.

[16] R. Ruotolo, C. Surace, Damage assessment of multiple cracked beams: numerical results and experimental validation. *Journal of Sound and Vibration* 206(4) (1997) 567-588.

[17] M.A. Rao, J. Srinivas, B.S.N. Murthy, Damage detection in vibrating bodies using genetic algorithms. *Computers and Structures* 82 (2004) 963-968.

[18] A. Teughels, Inverse modelling of civil engineering structures based on operational modal data. Phd Thesis, Katholieke Universitein, Leuven, Belgium, 2003.

[19] D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

[20] L. Ingberg, B. Rosen, Genetic algorithms and very fast simulated reannealing: a comparison. *Journal of Mathematical and Computer Modelling* 16(11) (1992) 87-100.

[21] J.A.K. Suykens, J. Vandewalle, B. De Moor, Intelligence and cooperative search by coupled local minimizers, *International Journal of Bifurcation and Chaos*. 11(8) (2001) 2133-2144.

[22] J.A.K., Suykens, J. Vandewalle, Coupled local minimizers: alternative formulations and extensions. *Proceeding of the World Congress on Computational Intelligence - International Joint Conference on Neural Networks IJCNN*, Honolulu, USA, 2002.

[24] R. Storn, K. Price, Differential Evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 11(4) (1997)

341-359.

[25] R. Storn, On the usage of differential evolution for function optimization. *Proceeding of NAFIPS 1996*, Berkeley, 1996, pp.519-523.

[26] M. Savoia, L. Vincenzi, Differential evolution algorithm for dynamic structural identification. *Proceedings of ICOSSAR 2005 conference*, Rome, Italy, June 19-23, 2005.

[27] M. Savoia, L. Vincenzi, Differential evolution algorithm for dynamic structural identification. *Journal of Earthquake Engineering* 12(5) (2008) 800-821.

[28] M. Savoia, B. Ferracuti, L. Vincenzi, Inverse analysis for calibration of FRP – concrete interface law. *Proceedings of FraMCoS 6 conference*, Catania, Italy, June 17-22, 2007.

[29] O. Hrstka, A. Kucerova, M. Leps, J. Zeman, A competitive comparison of different types of evolution algorithms. *Computers and Structures* 81 (2003)1979-1990.

[30] T.G. Chondros, A.D. Dimarogonas, J.Yao, A continuous cracked beam vibration theory. *Journal of Sound and Vibration* 215(1) (1998) 17-34.

[31] M. Breccolotti, A.L. Materazzi, I. Venanzi. Identification of the nonlinear behaviour of a cracked RC beam through the statistical analysis of the dynamic response. *Structural Control And Health Monitoring* 15 (2008): 416–435

[32] J. Nocedal, S.J. Wright, *Numerical optimization*. Springer, New York, USA, 1999.

[33] J. E. Mottershead, M.I. Friswell, Model updating in structural dynamics: a survey. *Journal of Sound and Vibration* 167(2) (1993) 347–375.

[34] N.M.M. Maia, J.M.M. Silva, J. He, *Theoretical and experimental modal analysis*. Research Studies Press, Somerset, England, 1997.

[35] J. Maeck, G. De Roeck, Damage detection on a prestressed concrete bridge and RC beams using dynamic system identification. *Proceedings of DAMAS 99*, Trans Tech Publications, Dublin, Ireland, 1999.

[36] R. Fox, M. Kapoor, Rate of change of eigenvalues and eigenvectors. *AIAA Journal* 6 (1968) 2426-2429.

[37] D.J. Ewins, *Modal Testing: Theory and Practice.* John Wiley & sons, New York,

2000.

[38] N. Hu, X.Wang, H. Fukunaga, Z.H. Yao, H.X. Zhang, Z.S. Wu, Damage assessment of structures using modal test data. *Journal of Solids and Structures* 38 (2001) 3111-3126.

[39] J.K. Sinha, M.I. Friswell, S. Edwards, Simplified models for the location of cracks in beam structures using measured vibration data. *Journal of Sound and Vibration* 251(1) (2002) 13-38.

[40] ANSYS, Robust simulation and analysis software. Available at: http://www.ansys.com, Release 10.0, ANSYS Incorporated, 2001.

[41] N.T. Khiem, T.V. Lien, A simplified method for natural frequency analysis of a multiple cracked beam. *Journal of Sound and Vibration* 245(4) (2001) 737-751.

[42] D.P. Patil, S.K. Maiti, Experimental verification of a method of detection of multiple cracks in beams based on frequency measurements. *Journal of Sound and Vibration* 281 (2005) 439-451.

List of Figure

List of Tables

DE algorithm: statistical analysis of identification results.

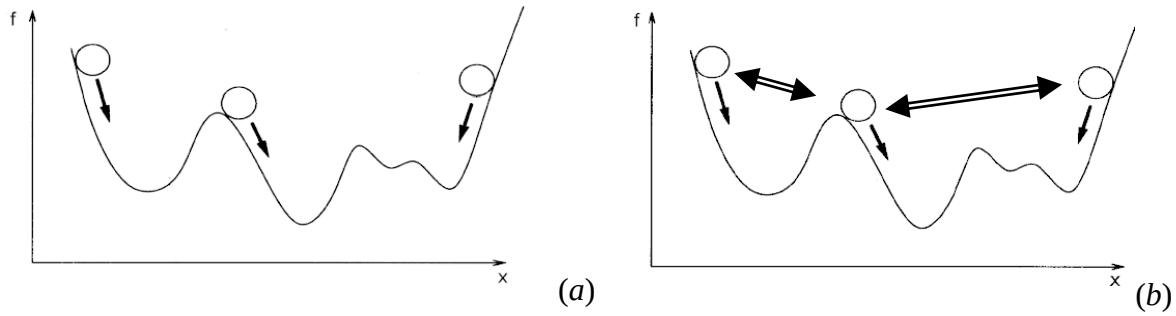
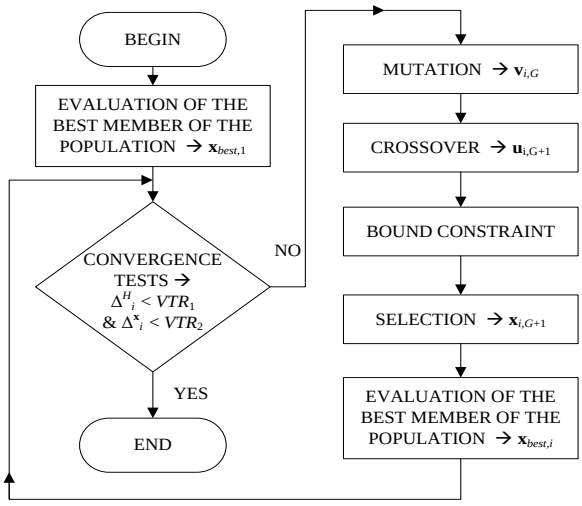
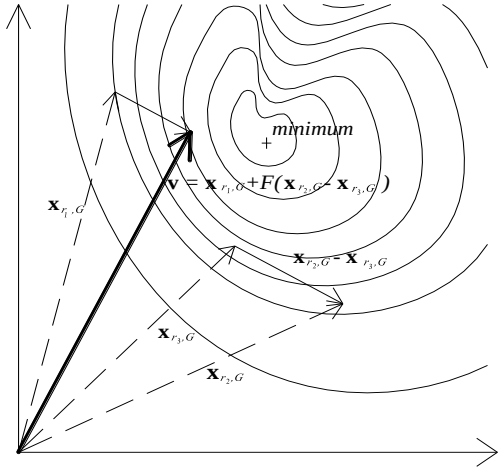


Fig. 1. Minimization of a cost function with more than one minimum: schemes of (*a*) independent multistart local optimization and (*b*) coupling introduced in the CLM method.

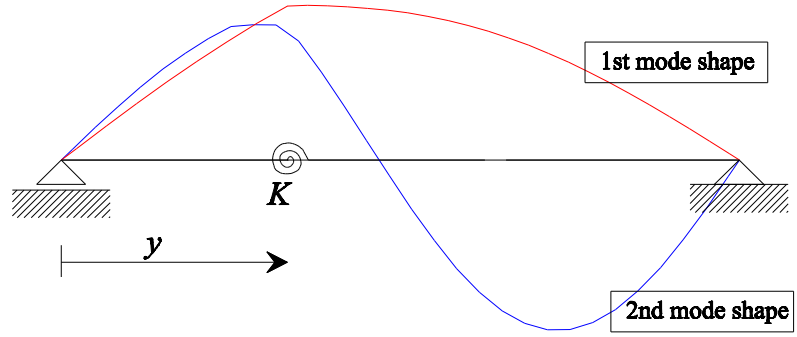


(*a*)

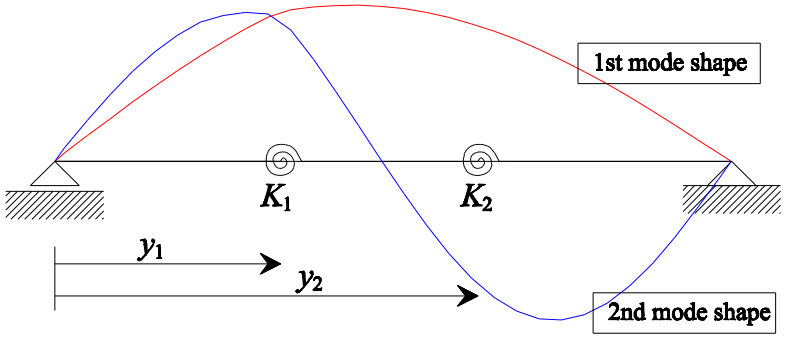


(*b*)

Fig. 2. (*a*) Algorithm scheme of DE method and (*b*) mutation process by random combination.

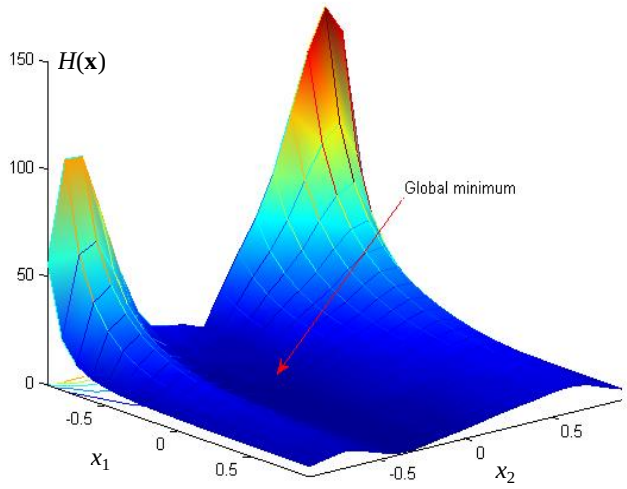


(*a*)

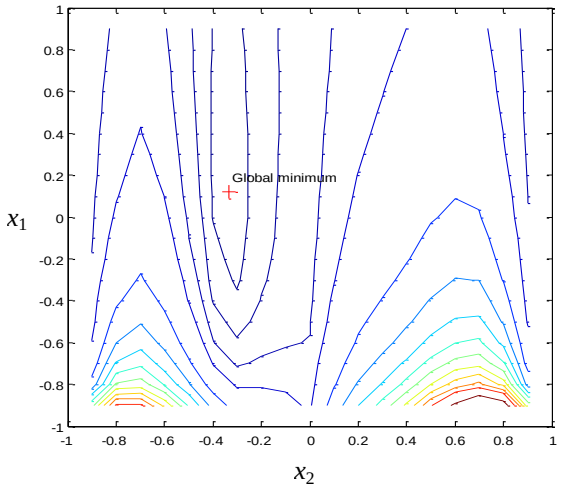


(*b*)

Fig. 3. The case studies - Geometry and first two mode shapes of (*a*) beam with one crack, (*b*) beam with two cracks, with position and depth to be determined.



(*a*)



(*b*)

Fig. 4. One-crack detection problem – the cost function: (*a*) 3D plot and (*b*) contour lines. x_1 : normalized spring stiffness, x_2 : normalized spring position.

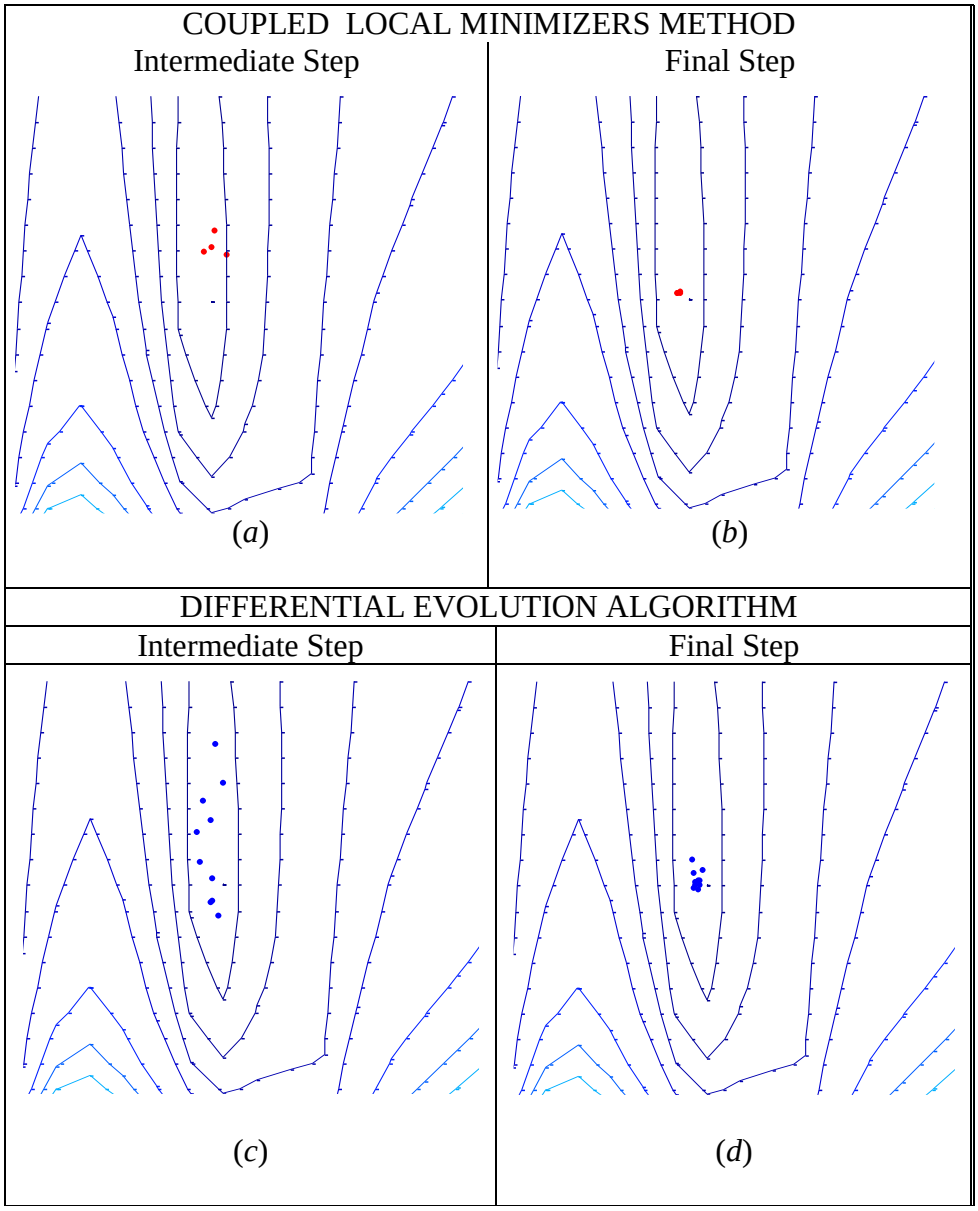


Fig. 5. One-crack detection problem: convergence of *(a,b)* CLM and *(c,d)* DE methods to the minimum of the cost function.

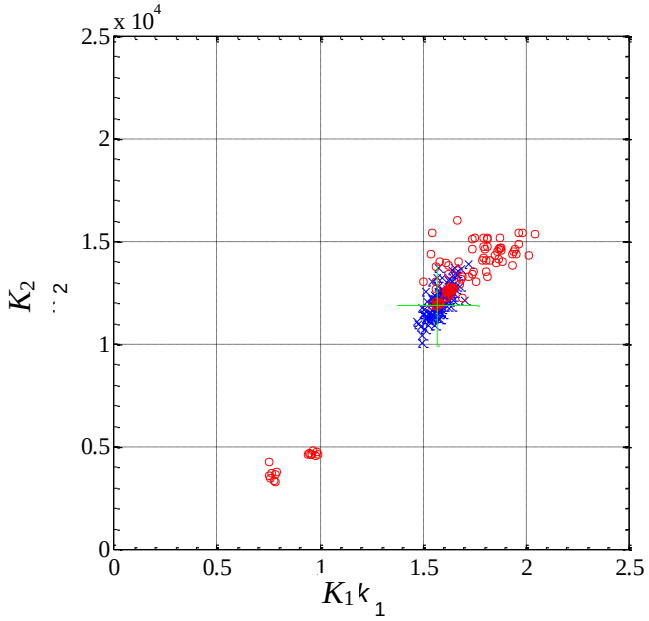


Fig. 6: Two-crack detection problem – Scatter of the identification results obtained by (x) DE and (o) CLM algorithms, adopting pseudo – experimental input data. +: the reference solution. In the plot, the two more scattered identification parameters (K_1, K_2) are reported.

Table 1

The reference solution of the one-crack problem: stiffness and position of the crack, first two flexural frequencies of the beam.

			Identification parameter	Normalized value
Stiffness (K)	x_1	[Nm]	15680	0.120
Position (y)	x_2	[mm]	78.3	0.333
Frequency	f_1	[Hz]	908.52	-
	f_2	[Hz]	3569.6	-

Table 2

One-crack detection problem, exact input data – Comparison between CLM and DE algorithm: statistical analysis of identification results from 100 tests.

		Reference values	**CLM**					**DE**				
			VALUE			*ERROR %*		*VALUE*			*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identified parameter	Stiffness K [Nm]	15680	15703	15625 ÷ 15819	0.28	0.25	0.88	15577	15363 ÷ 15730	0.63	0.70	2.02
	Position y [mm]	78.34	78.28	77.53 ÷ 78.94	0.39	0.30	1.03	78.26	78.16 ÷ 78.40	0.08	0.11	0.22
Cost function value:		-	0.0040	0.0004 ÷ 0.0204	-	-	-	0.0013	0.0005 ÷ 0.0033	-	-	-
Number of iterations:		-	14	10 ÷ 23	19	-	-	24	10 ÷ 40	35	-	-
Number of cost function evaluations:		-	174	120 ÷ 276	19	-	-	238	100 ÷ 400	35	-	-

Table 3

One-crack detection problem, exact input data – Comparison between CLM and DE algorithm: statistical analysis of identification results (adopting the frequency and 5 measure positions for each mode shape).

		Reference values	**CLM**					**DE**				
			VALUE			*ERROR %*		*VALUE*			*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identified parameter	Stiffness K [Nm]	15680	16154	14626 ÷ 17167	2.99	3.02	9.48	15762	15204 ÷ 19355	2.38	0.52	23.44
	Position y [mm]	78.34	78.21	75.00 ÷ 79.83	1.07	0.15	4.25	78.29	77.19 ÷ 78.49	0.25	0.05	1.46
Number of iterations:		-	16	8 ÷ 33	30	-	-	18	3 ÷ 26	22	-	-
Number of cost function evaluations:		-	192	96 ÷ 396	30	-	-	180	30 ÷ 260	22	-	-

Table 4

One-crack detection problem, pseudo-experimental input data – Comparison between CLM and DE algorithm: statistical analysis of identification results (adopting the frequency and 15 measure positions for each mode shape).

		Reference values	**CLM**					**DE**				
			VALUE			*ERROR %*		*VALUE*			*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identified parameter	Stiffness K [Nm]	15680	15792	15625 ÷ 15819	2.62	0.71	8.27	15730	14794 ÷ 16735	2.38	0.32	6.73
	Position y [mm]	78.34	78.35	74.19 ÷ 81.53	1.38	0.01	4.08	78.34	77.41 ÷ 79.09	0.42	0.00	0.96
Number of iterations:		-	15	6 ÷ 35	26	-	-	18	10 ÷ 26	17	-	-
Number of cost function evaluations:		-	180	72 ÷ 420	26	-	-	180	100 ÷ 260	17	-	-

Table 5

One-crack detection problem, pseudo-experimental input data – Comparison between CLM and DE algorithm: statistical analysis of identification results (adopting the frequency and 5 measure positions for each mode shape).

		Reference values	**CLM**					**DE**				
			VALUE			*ERROR %*		*VALUE*			*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identified parameter	Stiffness K [Nm]	15680	16861	14391 ÷ 19751	8.03	7.54	25.96	15768	14112 ÷ 17451	5.33	0.56	11.29
	Position y [mm]	78.34	78.00	75.67 ÷ 80.10	1.26	0.43	2.25	78.29	76.24 ÷ 79.76	0.94	0.06	1.81
Number of iterations:		-	17	9 ÷ 30	27	-	-	19	10 ÷ 27	16	-	-
Number of cost function evaluations:		-	204	108 ÷ 360	27	-	-	190	100 ÷ 270	16	-	-

Table 6

Two-cracks detection problem, exact input data – Comparison between CLM and DE algorithm.

		Reference value	**CLM**					**DE**				
				VALUE		*ERROR %*			*VALUE*		*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identif. parameter	Stiffness K_1 [Nm]	15680	16474	15762 ÷ 17051	2.48	5.06	8.74	15701	15418 ÷ 15940	0.74	0.57	1.67
	Position y_1 [mm]	78.34	78.45	75.81÷ 80.93	1.78	1.50	3.31	78.33	78.19 ÷ 78.47	0.08	0.05	0.20
	Stiffness K_2 [Nm]	11900	13151	12537 ÷ 13789	3.15	10.51	15.9	11933	11572 ÷ 12322	1.33	0.99	3.55
	Position y_2 [mm]	146.87	149.1	145.4 ÷ 153.5	1.34	1.71	4.53	146.94	146.49 ÷ 147.45	0.16	0.12	0.42
Cost function value:		-	0.0124	0.0010 ÷ 0.0750	-	-	-	$1.6\cdot10^{-4}$	$0.1\cdot10^{-4}$÷ $5.6\cdot10^{-4}$	-	-	-
Number of iterations:		-	15.68	9 ÷ 32	44.7	-	-	29.72	24 ÷ 37	10.4	-	-
Number of cost function evaluations:		-	784	450 ÷1600	44.7	-	-	743	600 ÷ 925	10.4	-	-

Table 7

Two-crack detection problem, pseudo-experimental input data – Comparison between CLM and DE algorithm: statistical analysis of identification results.

		Reference value	**CLM**					**DE**				
				VALUE		*ERROR %*			*VALUE*		*ERROR %*	
			Mean	Range	C.V.%	Mean	Max	Mean	Range	C.V.%	Mean	Max
Identif. parameter	Stiffness K_1 [Nm]	15680	15953	8506 ÷ 23195	21.54	1.74	47.92	15828	14752 ÷ 17234	3.26	0.95	9.91
	Position y_1 [mm]	78.34	78.75	69.42.÷ 87.24	4.53	0.53	11.37	78.32	77.57 ÷ 79.00	0.38	0.01	0.96
	Stiffness K_2 [Nm]	11900	12324	4114 ÷ 19849	30.02	3.56	66.79	12115	10071 ÷ 13901	6.37	1.81	16.81
	Position y_2 [mm]	146.87	153.26	133.48 ÷ 171.49	5.46	4.35	16.76	147.22	144.31 ÷ 150.43	0.83	0.24	2.42
Number of iterations:		-	183	30 ÷ 1172	112	-	-	35	25 ÷ 50	14.9	-	-
Number of cost function evaluations:		-	3660	600 ÷ 23440	112	-	-	875	625 ÷ 1250	14.9	-	-